| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/javax/swing/text/GlyphView.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text//class-useGlyphView.html)    [**NO FRAMES**](http://docs.google.com/GlyphView.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Class**

**javax.swing.text.GlyphView**

| Packages that use [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) | |
| --- | --- |
| [**javax.swing.text**](#3znysh7) | Provides classes and interfaces that deal with editable and noneditable text components. |
| [**javax.swing.text.html**](#2et92p0) | Provides the class HTMLEditorKit and supporting classes for creating HTML text editors. |

| Uses of [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |

| Subclasses of [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |
| class | [**LabelView**](http://docs.google.com/javax/swing/text/LabelView.html)            A LabelView is a styled chunk of text that represents a view mapped over an element in the text model. |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) with parameters of type [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) | |
| --- | --- |
| abstract  float | **GlyphView.GlyphPainter.**[**getAscent**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getAscent(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| abstract  int | **GlyphView.GlyphPainter.**[**getBoundedPosition**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getBoundedPosition(javax.swing.text.GlyphView,%20int,%20float,%20float))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, float x, float len)            Determines the model location that represents the maximum advance that fits within the given span. |
| abstract  float | **GlyphView.GlyphPainter.**[**getDescent**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getDescent(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| abstract  float | **GlyphView.GlyphPainter.**[**getHeight**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getHeight(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| int | **GlyphView.GlyphPainter.**[**getNextVisualPositionFrom**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getNextVisualPositionFrom(javax.swing.text.GlyphView,%20int,%20javax.swing.text.Position.Bias,%20java.awt.Shape,%20int,%20javax.swing.text.Position.Bias%5B%5D))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int pos, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) b, [Shape](http://docs.google.com/java/awt/Shape.html) a, int direction, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasRet)            Provides a way to determine the next visually represented model location that one might place a caret. |
| [GlyphView.GlyphPainter](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html) | **GlyphView.GlyphPainter.**[**getPainter**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getPainter(javax.swing.text.GlyphView,%20int,%20int))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, int p1)            Create a painter to use for the given GlyphView. |
| abstract  float | **GlyphView.GlyphPainter.**[**getSpan**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getSpan(javax.swing.text.GlyphView,%20int,%20int,%20javax.swing.text.TabExpander,%20float))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, int p1, [TabExpander](http://docs.google.com/javax/swing/text/TabExpander.html) e, float x)            Determine the span the glyphs given a start location (for tab expansion). |
| abstract  [Shape](http://docs.google.com/java/awt/Shape.html) | **GlyphView.GlyphPainter.**[**modelToView**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#modelToView(javax.swing.text.GlyphView,%20int,%20javax.swing.text.Position.Bias,%20java.awt.Shape))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int pos, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias, [Shape](http://docs.google.com/java/awt/Shape.html) a)            Provides a mapping from the document model coordinate space to the coordinate space of the view mapped to it. |
| abstract  void | **GlyphView.GlyphPainter.**[**paint**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#paint(javax.swing.text.GlyphView,%20java.awt.Graphics,%20java.awt.Shape,%20int,%20int))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, [Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Shape](http://docs.google.com/java/awt/Shape.html) a, int p0, int p1)            Paint the glyphs representing the given range. |
| abstract  int | **GlyphView.GlyphPainter.**[**viewToModel**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#viewToModel(javax.swing.text.GlyphView,%20float,%20float,%20java.awt.Shape,%20javax.swing.text.Position.Bias%5B%5D))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, float x, float y, [Shape](http://docs.google.com/java/awt/Shape.html) a, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasReturn)            Provides a mapping from the view coordinate space to the logical coordinate space of the model. |

| Uses of [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) | |
| --- | --- |

| Subclasses of [GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) | |
| --- | --- |
| class | [**InlineView**](http://docs.google.com/javax/swing/text/html/InlineView.html)            Displays the inline element styles based upon css attributes. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/javax/swing/text/GlyphView.html) | **Use** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text//class-useGlyphView.html)    [**NO FRAMES**](http://docs.google.com/GlyphView.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).